



(19)

(11) Publication number:

63047867 A

Generated Document

## PATENT ABSTRACTS OF JAPAN

(21) Application number: 61191935

(51) Intl. Cl.: G06F 15/16 G06F 13/38

(22) Application date: 16.08.86

(30) Priority:

(43) Date of application publication: 29.02.88

(84) Designated contracting states:

(71) Applicant: NEC CORP

(72) Inventor: OTANI TSUTOMU

(74) Representative:

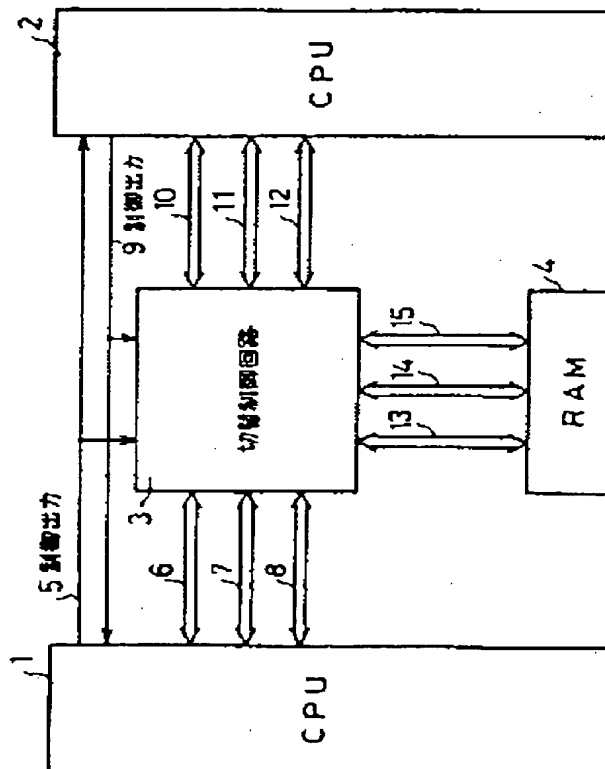
## (54) INTER-DUAL CPU COMMUNICATION SYSTEM

(57) Abstract:

**PURPOSE:** To attain the transmission/reception of a large capacity of data at a high speed and low cost, by writing a transmission data on a RAM in parallel through a data bus, by a CPU on a transmission side, and reading out a written data by the CPU on a reception side.

**CONSTITUTION:** A first CPU, when the transmission data being generated, confirms the nonuse of the RAM4 by a second CPU2 based on whether the controlled output 9 of the second CPU2 becomes active or not, and outputs the transmission data to the data bus 8. In this way, the transmission data from the CPU1 is written on the RAM4 through a switching control circuit 3. Also, the second CPU2 confirms the nonuse of the RAM4 by the first CPU1 based on whether the controlled output 5 of the CPU1 becomes active or not, and reads the transmission data from the CPU1 stored in the RAM4.

COPYRIGHT: (C)1988,JPO&amp;Japio



⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A)

昭63-47867

⑮ Int. Cl.<sup>1</sup>

識別記号

庁内整理番号

⑭ 公開 昭和63年(1988)2月29日

G 06 F 15/16  
13/38  
15/16

4 0 0  
3 3 0  
4 0 0

M-2116-5B  
F-7165-5B  
B-2116-5B

審査請求 未請求 発明の数 1 (全6頁)

⑬ 発明の名称 デュアルCPU間通信方式

⑯ 特 願 昭61-191935

⑰ 出 願 昭61(1986)8月16日

⑱ 発 明 者 大 谷 努 東京都港区芝5丁目33番1号 日本電気株式会社内  
⑲ 出 願 人 日本電気株式会社 東京都港区芝5丁目33番1号  
⑳ 代 理 人 弁理士 境 廣 巳

#### 明 細 書

##### 1. 発明の名称

デュアルCPU間通信方式

##### 2. 特許請求の範囲

第1、第2のCPUが互いにデータの送受信を行ないながら処理を分担するデュアルCPU回路に於いて、

RAMと、

前記第1、第2のCPUのアドレスバス、データバス及びステータスバスを前記RAMに接続する可否を制御する切替制御回路とを有し、

前記第1、第2のCPUはデータ送信時、他方のCPUの制御出力がアクティブであるか否かを判断し、アクティブでないと判断することにより自己の制御出力をアクティブにすると共に再度前記他方のCPUの制御出力がアクティブであるか否かを判断し、再度アクティブでないと判断することにより前記アドレスバス、データバス及びステータスバスを用いて前記RAMに送信データを書込み、

前記第1、第2のCPUはデータ受信時、他方のCPUの制御出力がアクティブであるか否かを判断し、アクティブでないと判断することにより自己の制御出力をアクティブにすると共に再度前記他方のCPUの制御出力がアクティブであるか否かを判断し、再度アクティブでないと判断することにより前記アドレスバス、データバス及びステータスバスを用いて前記RAMに書込まれている受信データを読出し、

前記切替制御回路は前記第1、第2のCPUのアドレスバス、データバス及びステータスバスの内、制御出力がアクティブになっているCPUのアドレスバス、データバス及びステータスバスを前記RAMに接続することを特徴とするデュアルCPU間通信方式。

##### 3. 発明の詳細な説明

(産業上の利用分野)

本発明は2個のCPU(プロセッサ)が互いにデータの送受信を行ないながら処理を分担するデュアルCPU回路に関し、特にCPU間の通信方

式に関する。

(従来の技術)

従来、この種のデュアルCPU間通信方式は、CPUのシリアルポートを用いて直列にデータを送受信する方式、或いはCPU間にデータバスを介してFIFO(First-in-first-out)を配置し、並列にデータを送受信する方式となっている。

(発明が解決しようとする問題点)

上述した従来のCPU間通信方式に於いて、CPUシリアル・ポートを用いて直列にデータを送受信する方式は転送速度が限られるので、大容量のデータを高速に送受信できない問題があった。また、CPU間にデータバスを介してFIFOを配置し、並列にデータを送受信する方式は、FIFOの段数が限られるため、大容量のデータを一括して送受信できない問題があると共にビットコストが高くなる問題があった。

本発明は前述の如き問題点を解決したものであり、その目的は、低ビットコストで大容量のデータを高速に送受信できるようにすることにある。

クティブにすると共に再度前記他方のCPUの制御出力がアクティブであるか否かを判断し、再度アクティブでないと判断することにより前記アドレスバス、データバス及びステータスバスを用いて前記RAMに書込まれている受信データを読出す構成を有し、

前記切替制御回路は前記第1、第2のCPUのアドレスバス、データバス及びステータスバスの内、制御出力がアクティブになっているCPUのアドレスバス、データバス及びステータスバスを前記RAMに接続する構成を有する。

(作用)

データの送受信は送信側のCPUがデータバスを介して並列的にRAMに送信データを書込み、受信側のCPUがRAMに書込まれたデータを読出すことにより行なわれる。従って、高速に大容量のデータを送受信することが可能となる。また、CPUはデータを送受信する場合、即ちRAMをアクセスする場合、相手方のCPUの制御出力がアクティブになっているかを判断し、アクティブ

(問題点を解決するための手段)

本発明は前述の如き問題点を解決するため、

第1、第2のCPUが互いにデータの送受信を行ないながら処理を分担するデュアルCPU回路に於いて、

RAMと、

前記第1、第2のCPUのアドレスバス、データバス及びステータスバスを前記RAMに接続するか否かを制御する切替制御回路とを設け、

前記第1、第2のCPUはデータ送信時、他方のCPUの制御出力がアクティブであるか否かを判断し、アクティブでないと判断することにより自己の制御出力をアクティブにすると共に再度前記他方のCPUの制御出力がアクティブであるか否かを判断し、再度アクティブでないと判断することにより前記アドレスバス、データバス及びステータスバスを用いて前記RAMに送信データを書込み、データ受信時、他方のCPUの制御出力がアクティブであるか否かを判断し、アクティブでないと判断することにより自己の制御出力をア

クティブにすると共に再度前記他方のCPUの制御出力がアクティブであるか否かを判断し、再度アクティブでないと判断することにより前記アドレスバス、データバス及びステータスバスを用いて前記RAMに書込まれている受信データを読出す構成を有し、

(実施例)

次に本発明の実施例について図面を参照して説明する。

第1図は本発明の実施例のブロック図であり、RAM 4をアクセスする際、制御出力5をアクティブ("1")とする第1のCPU 1と、RAM 4をアクセスする際、制御出力9をアクティブとする第2のCPU 2と、切替制御回路3とから構成されている。尚、RAM 4はCPU 1からCPU 2にデータを転送する際に使用する第1の領域と、CPU 2からCPU 1にデータを転送する際に使用する第2の領域とを有しているものである。また、同図に於いて、6、10、13はアドレスバス、

7, 11, 14はステータスバス、8, 12, 15はデータバスである。

また、第2図は切替制御回路3の構成例を示すブロック図であり、制御信号が“1”となることによりアクティブ状態となるスリーステートバッファB1～B8と、制御出力5を一方の入力としステータスバス7の所定ビットを他方の入力とするアンドゲートA1と、制御出力9を一方の入力としステータスバス11の所定ビットを他方の入力とするアンドゲートA2と、制御出力5を一方の入力としステータスバス7の所定ビットを他方の入力とするインヒビットゲートI1と、制御出力9を一方の入力としステータスバス11の所定ビットを他方の入力とするインヒビットゲートI2とから構成されている。尚、ステータスバス7, 11の前記所定ビットはRAM4にデータを書込む場合には“1”となり、RAM4からデータを読み出す場合には“0”となるものである。従って、CPU1がRAM4にデータを書込む場合はスリーステートバッファB1, B3, B5がアクティブ

状態となり、CPU2がデータを書込む場合はスリーステートバッファB2, B4, B7がアクティブ状態となり、CPU1がRAM4からデータを読み出す場合はスリーステートバッファB1, B3, B6がアクティブ状態となり、CPU2がRAM4からデータを読み出す場合はスリーステートバッファB2, B4, B8がアクティブ状態となる。

次に、第1のCPU1から第2のCPU2にデータを送信する場合を例にとって、動作を説明する。

第1のCPU1は第3図のフローチャートに示すように、送信データが発生すると(ステップS1)、第2のCPU2がRAM4を使用しているか否かをCPU2の制御出力9がアクティブになっているか否かに基づいて判断する(ステップS2)。そして、第2のCPU2がRAM4を使用していないことを確認すると、第1のCPU1は自己の制御出力5をアクティブとする(ステップS3)。次いで、第1のCPU1は第2のCPU

2とのRAM4に対する使用要求の衝突を防止するため、再度CPU2の制御出力9を確認し(ステップS4)、制御出力9がノンアクティブであれば、アドレスバス6に書込アドレスを出力し、ステータスバス7に書込みを行なうことを示すステータス情報を出力し、データバス8に送信データを出力する(ステップS5)。これにより、切替制御回路3内のスリーステートバッファB1, B3, B5のみがアクティブ状態となり、アドレスバス6とアドレスバス13とが接続され、ステータスバス7とステータスバス14とが接続され、データバス8とデータバス15とが接続されるので、第1のCPU1からの送信データがRAM4の第1の領域に書込まれる。そして、送信データの送出が完了すると、CPU1は第1の領域の所定番地#Aにフラグとして“1”を記憶させ(ステップS6)、この後、他の制御ステップに移る。尚、ステップS2の判断結果がYESの場合、即ち、第2のCPU2がRAM4をアクセスしている場合は、一定時間の待合せ処理を行なった後(ステッ

プS8)、再度ステップS2の処理を行ない、ステップS4の判断結果がYESの場合は、制御出力5をノンアクティブにし(ステップS7)、一定時間の待合せ処理を行なった後(ステップS8)、再度ステップS2の処理を実行する。

また、第2のCPU2は第4図のフローチャートに示すように、受信データの有無をチェックするタイミングとなると(ステップS11)、第1のCPU1がRAM4を使用しているか否かを制御出力5がアクティブになっているか否かによって判断する(ステップS12)。そして、第1のCPU1がRAM4を使用していないことを確認すると、第2のCPU2は自己の制御出力9をアクティブとする(ステップS13)。次いで、CPU2はCPU1とのRAM4に対する使用要求の衝突を防止するため、再度CPU1の制御出力5を確認し(ステップS14)、ノンアクティブであれば、受信データがあるか否かをRAM4の前記第1の領域の所定番地#Aに記憶されているフラグFが“1”になっているか否かによって判断する(ス

テップS15)。

そして、受信データ有りと判断した場合は、CPU2はアドレスバス10に読出アドレスを出力し、ステータスバス11に読出しを行なうことを示すステータス情報を出力する(ステップS16)。これにより、切替制御回路3内のスリーステートバッファB2、B4、B8がアクティブ状態になるので、RAM4の第1の領域に記憶されていた第1のCPU1からの送信データが第2のCPU2に読込まれることになる。そして、CPU1からの送信データを全て読込むと、即ち受信が完了すると、CPU2はRAM4の第1の領域の所定番地#Aに記憶されているフラグFを"0"とし(ステップS17)、この後、他の制御ステップに移る。また、ステップS12の判断結果がYESの場合、即ち、CPU1がRAM4をアクセスしている場合は、一定時間の待合せ処理を行なった後(ステップS19)、再度ステップS2の処理を行ない、ステップS14の判断結果がYESの場合は制御出力9をノンアクティブに戻し(ステップS18)、一定

時間の待合せ処理を行なった後(ステップS19)、再度ステップS12の処理を実行する。

尚、上述した実施例は、第1のCPU1から第2のCPU2へデータを送出する場合について説明したが、第2のCPU2から第1のCPU1にデータを送出する場合も同様な動作が行なわれるものである。また、上述した実施例に於いては、RAM4をアクセスする際、相手方CPUの制御出力を2回見るようにしたが、複数回であれば、これに限られるものではない。

#### (発明の効果)

以上説明したように、本発明は、送信側のCPUがデータベースを介して並列的にRAMに送信データを書込み、受信側のCPUがRAMに書込まれたデータを読出すことにより、CPU間でデータの送受信を行なうものであるから、大容量のデータを高速且つ低いビットコストで送受信することができる利点がある。また、更に、本発明は、RAMをアクセスする際、相手方のCPUの制御出力がアクティブになっているか否かを判断し、

アクティブでないと判断することにより自CPUの制御出力をアクティブにすると共に再度相手方の制御出力がアクティブになっているか否かを判断し、再度アクティブになっていないと判断することによりRAMをアクセスするものであるから、第1、第2のCPUが同時にRAMをアクセスすることを防止できる利点もある。

#### 4. 図面の簡単な説明

第1図は本発明の実施例のブロック図、

第2図は切替制御回路の構成例を示すブロック図、

第3図はCPU1の処理内容の一部を示すフローチャート及び、

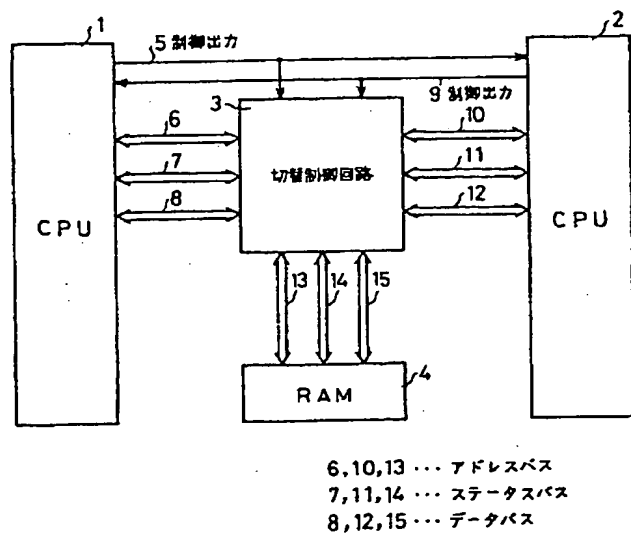
第4図はCPU2の処理内容の一部を示すフローチャートである。

図に於いて、1、2…CPU、3…切替制御回路、4…RAM、6、10、13…アドレスバス、7、11、14…ステータスバス、8、12、15…データベース、B1～B8…スリーステートバッファ、A1、A2…アンドゲート、I1、I2…インヒビット

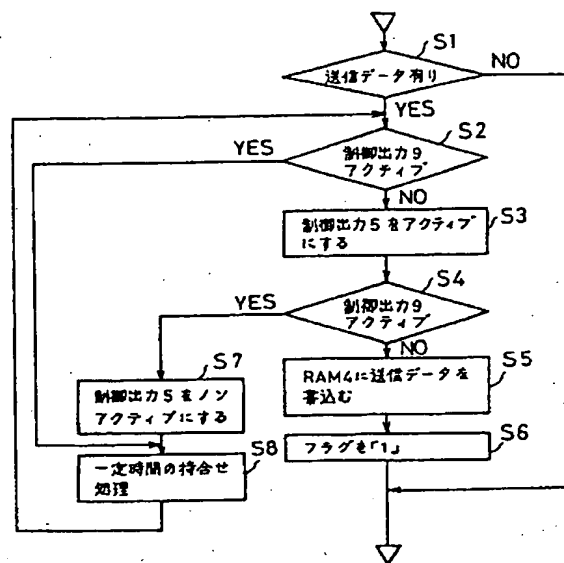
ゲート。

特許出願人 日本電気株式会社

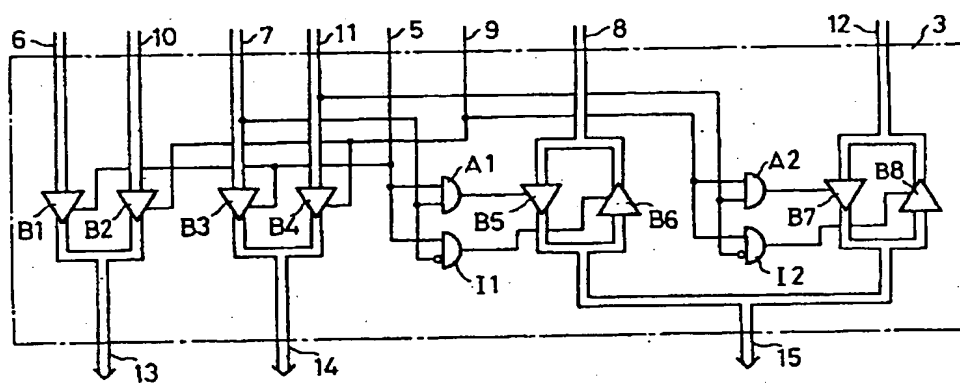
代理人 弁理士 境 廣 巳



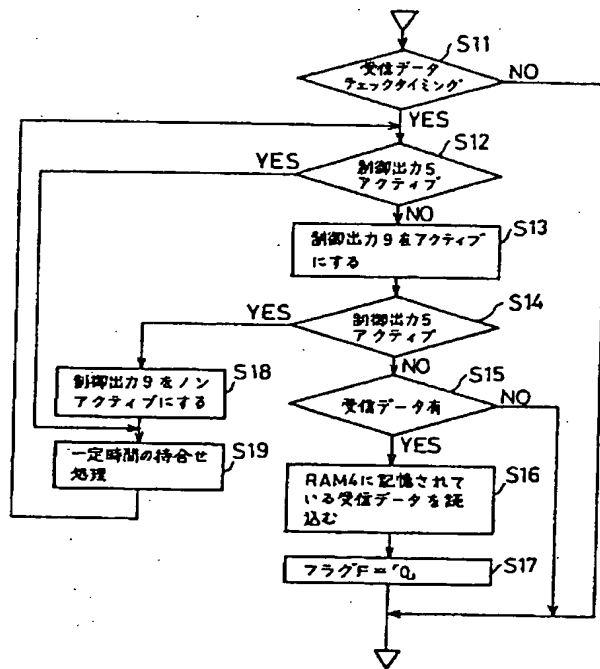
本発明の実施例のブロック図  
第 1 図



送信時の処理内容の一部を示すフローチャート  
第 3 図



切替制御回路の構成例を示すブロック図  
第 2 図



受信時の処理内容の一部を示すフローチャート

第4図